IMPLEMENTASI *PORT KNOCKING* PADA SISTEM KEAMANAN JARINGAN DENGAN MENERAPKAN ALGORITMA RSA (*RIVEST SHAMIR ADLEMAN*)

SKRIPSI

Skripsi ini sebagai salah satu syarat untuk memperoleh gelar sarjana Teknik Informatika
Universitas Pembangunan Nasional ”Veteran” Yogyakarta

Disusun oleh :

I KOMANG HARTAWAN WIJAYA
123060051

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL ”VETERAN”
YOGYAKARTA
2011

ABSTRAK

Saat ini keamanan telah menjadi hal yang sangat penting, terutama dalam bidang Teknologi Informasi. Statistik tingkat eksploitasi keamanan terhadap banyak *server* dan jaringan makin hari semakin meningkat. Bahkan seseorang yang tidak memiliki pengetahuan yang cukup dalam masalah keamanan jaringan dapat melakukan penetrasi terhadap sebuah sistem jaringan dengan hanya men-*download exploit* untuk sistem yang diserangnya dan kemudian menggunakannya untuk kepentingan sendiri.

Salah satu upaya mengamankan sebuah *server* adalah dengan menggunakan *firewall*, tetapi saat ini *firewall* masih memiliki kelemahan. Sehingga dicari solusi terbaik agar dapat mengakses *service* tertentu walaupun port tersebut tertutup. Metode untuk membuka port tertutup secara eksternal tersebut bernama metode *port knocking*. Metode yang digunakan dalam membangun aplikasi ini adalah dengan Metode *Waterfall*, dengan penyesuaian yang diperlukan untuk membangun aplikasi ini, meliputi rekayasa dan permodelan sistem, analisa kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Dalam pembuatan aplikasi ini menggunakan *Netbeans IDE 2.9.1* dan *CHX packet filter 3.0* sebagai *firewall*.

Berdasarkan hasil analisis dan perancangan, telah berhasil dibuat aplikasi port knocking yang dapat melakukan koneksi ke *server* meskipun port yang dituju tertutup oleh *firewall*. Di dalam uji coba telah berhasil dilakukan akses ke beberapa port *service* yang tertutup seperti port FTP(21), SSH(22), TELNET(23), HTTP(80). Penelitian ini diharapkan dapat membantu administrator untuk mengelola *server* dari *remote area* dan meminimalisir serangan terhadap *server* dengan menyembunyikan *service* yang sedang berjalan pada port tertentu.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR MODUL